I-Louvain: An Attributed Graph Clustering Method

David Combe, Christine Largeron, Mathias Géry, Elod Egyed-Zsigmond

I-Louvain: an attributed graph clustering method

David Combe[1], Christine Largeron[1], Mathias Géry[1], and Előd Egyed-Zsigmond[2]

[1] Université de Lyon, F-42023, Saint-Étienne, France,
CNRS, UMR 5516, Laboratoire Hubert Curien, F-42000, Saint-Étienne, France
Université de Saint-Étienne, Jean-Monnet, F-42000, Saint-Étienne, France
{ david.combe, christine.largeron, mathias.gery
}@univ-st-etienne.fr
[2] Université de Lyon, UMR 5205 CNRS, LIRIS
7 av J. Capelle, F-69100 Villeurbanne, France
elod.egyed-zsigmond@insa-lyon.fr

Abstract. Modularity allows to estimate the quality of a partition into communities of a graph composed of highly inter-connected vertices. In this article, we introduce a complementary measure, based on inertia, and specially conceived to evaluate the quality of a partition based on real attributes describing the vertices. We propose also **I-Louvain**, a graph nodes clustering method which uses our criterion, combined with Newman's modularity, in order to detect communities in attributed graph where real attributes are associated with the vertices. Our experiments show that combining the relational information with the attributes allows to detect the communities more efficiently than using only one type of information. In addition, our method is more robust to data degradation.

Keywords: Attributed graph, Graph clustering, Social network, Community detection, Modularity

1 Introduction

Clustering of graph vertices is a task related to community detection within social networks. The goal is to create a partition of the vertices, taking into account the topological structure of the graph, in such a way that the clusters are composed of strongly connected vertices [13, 23, 29, 20, 3]. Among the core methods proposed in the literature, we can cite those that optimize a function (modularity, ratio cut or its variants, etc.) in order to evaluate the quality of the partition [19, 30, 10, 25], the hierarchical techniques like divisive algorithms based on the minimum cut [14], the spectral methods [34] or the Markov Clustering algorithm and its extensions [28]. We refer to the survey of Fortunato for a thorough discussion of community detection methods [15].

Graph clustering techniques are very useful for detecting strongly connected groups in a graph but many of them mainly focus on the topological structure, ignoring the vertices properties. Nowadays, various data sources can be seen as graphs where vertices have attributes and a new challenge in graph clustering consists in combining the relational information corresponding to the network and attributes describing the vertices. Generally, this is not the case in clustering of vertices where only the relationships between the vertices are used, nor in unsupervised classification based only

on the attributes. Recently, several methods have been proposed to take into account the relational information as well as the attributes in the aim to detect patterns in attributed graphs [26, 31] or to tackle this problem of hybrid clustering [6, 11]. In this article, we propose a method, called **I-Louvain**, which allows to partition the vertices of an attributed graph when numerical attributes are associated to the vertices. In social networks, these attributes can correspond to features (age or weight) or *tf-idf* vector representing documents associated to the nodes. This method is based on a local optimization of a global criterion which is a function on the one hand of the modularity [24] and on the other hand of a new measure based on inertia.

After a presentation of related work in section 2, we define this measure, called inertia based modularity, in section 3, and the method **I-Louvain** in section 4. The experimental study of section 5 confirms that clustering, based on the relational information and attributes provides more meaningful clusters than methods taking into account one type of data (attributes or edges) or than ToTeM which exploits attributes and edges [6].

2 Related work

Recently, methods exploiting both information types were introduced in order to detect communities in social networks or graphs where vertices have attributes.

Steinhaeuser and Chawla propose to measure the similarity between vertices according to their attributes and then to use the result as a weight of the edge linking the two vertices. After this pre-treatment, they use a graph partitioning method in order to cluster the new weighted graph [32]. In the hierarchical clustering of Li *et al.*, after a first phase consisting in detecting community seeds with the relational information, the final communities are built under constraints defined by the attributes [21]. This leads to merging the seeds on the base of their attributes' similarity. So, in these previous methods, the two types of information are not exploited simultaneously.

Zhou *et al.* exploit the attributes in order to extend the original graph [36, 37]. They add new vertices representing the attributes and new edges that link original vertices having similar attributes through these new vertices. A graph partitioning is then carried out on this new augmented graph. However, this approach cannot be used when the attributes have continuous values: it works only with categorical attributes.

Ester *et al.* study the "connected k-center problem" and propose a method called **NetScan**, which is an extended version of the **K-means** algorithm with an internal connectivity constraint [12, 16]. Under this constraint, two vertices in a same cluster are connected by a path that is internal to the cluster. In NetScan as in many other partitioning methods, the number of clusters has to be known in advance. However, this condition is relaxed in the work of Moser [22].

CESNA was introduced by Yang *et al.* to identify Communities from Edge Structure and Node Attributes [35]. One advantage of this method is its ability to detect overlapping communities by modeling the interaction between the network structure and the node attributes.

There are some other methods, focusing on dense subgraph detection, that integrate the homogeneity of the attributes inside the subgraphs, cf. for instance [17, 18].

Finally, we can mention a family of methods which propose to extend the well-known **Louvain** algorithm and for this reason, they are probably the most related works to our concerns. Dang *et al.* suggest to modify the modularity by considering not only the link between two vertices but also the similarity of their attributes. Thus, the two types of information are simultaneously considered in the partitioning process but with this approach, the communities provided can contain non linked vertices [9]. In [7], the optimization phase of the **Louvain** algorithm is based not only on the modularity but also on the entropy of the partition but, again, the two types of information are not exploited simultaneously.

Recently, some of these methods have been compared and these experiments have confirmed that the detection of communities in an attributed graph is not a trivial problem [6, 11]. To solve it efficiently, we consider that the attributes and the relational information must be exploited simultaneously and this is not the case for several methods cited. Moreover, the majority of the methods quoted previously exploit categorical attributes but they are not suited for numerical attributes. This is the reason for which, in this article, we propose **I-Louvain**, a method to detect communities in a graph where numerical attributes are associated to the vertices. These attributes can correspond to features (age or weight) or to a *tf-idf* vector representing documents associated to the vertex. **I-Louvain** consists in optimizing on the one hand the modularity introduced by Newman [24] and on the other hand a new measure that is defined in the next section.

3 Inertia based modularity

Let V be a set of N elements represented in a real vector space such that each element $v \in V$ is described by a vector of attributes $v = (v_1, \ldots, v_{|T|}) \in \mathbb{R}^{|T|}$. The inertia $I(V)$ of V through its center of gravity g, also called second central moment, is an homogeneity measure defined by $I(V) = \sum_{v \in V} \|v - g\|^2$, where $\|v' - v\|$ denotes the euclidean distance between v and v', $g = (g_1, \ldots, g_{|T|})$, the center of gravity of V is such that $g_j = \frac{1}{N} \sum_{v \in V} v_j$.

The inertia $I(V, v)$ of V through v is equal to the sum of the square euclidean distances between v and the other elements of V: $I(V, v) = \sum_{v' \in V} \|v' - v\|^2$

Given a partition $\mathscr{P} = \{C_1, \ldots, C_r\}$ of V in r disjoint clusters, we introduce a quality measure $Q_{inertia}(\mathscr{P})$ of $\mathscr{P}$ defined by:

$$Q_{inertia}(\mathscr{P}) = \sum_{(v,v') \in V \cdot V} \left[\left(\frac{I(V,v) \cdot I(V,v')}{(2N \cdot I(V))^2} - \frac{\|v - v'\|^2}{2N \cdot I(V)} \right) \cdot \delta\left(c_v, c_{v'}\right) \right] \quad (1)$$

where c_v denotes the cluster of $v \in V$ and δ is the Kronecker function equal to 1 if c_v and $c_{v'}$ are equal and 0 otherwise.

Thus, while the modularity, introduced by Newman, considers the strength of the link between vertices in order to cluster strongly connected vertices, our measure attempts to cluster elements which are the most similar. This appears in the second term of the equation 1, which is a function of the square of the distance between v and v',

corresponding to an *observed* distance between v and v'. This *observed* distance between v and v' is compared with an *expected* distance deducted from their respective inertia. This *expected* distance, which appears in the second term of the equation 1, is a function of the square distance of each of these elements v and v' to the other elements of V.

Therefore, $Q_{inertia}$ allows to compare, for each pair of elements (v, v') from the same community, the *expected* distance with the *observed* distance. If the former is greater than the latter, then v and v' are good candidates to be affected in a same cluster.

Given the normalization factors in the denominators of the expected and observed distances, the criterion $Q_{inertia}$ ranges between -1 and 1. Indeed, the maximum value of the left term in the subtraction (Eq. 1), containing the product of the inertia for all pairs of elements is 1. Similarly, the right term of the criterion $Q_{inertia}$ (Eq. 1) can not exceed 1. Both terms are strictly positive. Consequently the measure, constrained by the Kronecker function, varies between -1 and 1.

This criterion has several interesting properties. Firstly, it has the same value irrespective of the affine transformation applied to the attribute vectors, in other words the addition of a constant and / or the multiplication by a scalar of the vectors associated to the elements do not affect the value $Q_{inertia}$. Secondly, the order of attributes has no effect on the result.

However, this criterion has also limitations. It is undefined if the vectors are identical, since the total inertia is then zero. This is not really a problem, because in this case, the detection of the communities will be based only on the relational data. Moreover, as the modularity introduced by Newman, this criterion could present a resolution limit. If it is the case, the solution proposed by Arenas *et al.* or Reichardt and *et al.* could be adapted for our criterion [1, 27].

4 I-Louvain

As stated above, a direct application of our measure $Q_{inertia}$ is the community detection in social networks represented by an attributed graph $G = (V, E)$ where V is a set of vertices, E is a set of edges and where each vertex $v \in V$ is described by a real attribute vector $v = (v_1, \ldots, v_j, \ldots, v_T) \in \mathbb{R}^{|T|}$ [36]. In this section, we propose a community detection method for real attributed graphs which exploits the inertia-based modularity $Q_{inertia}$ jointly with the Newman modularity $Q_{NG}(\mathscr{P})$. Our method, called **I-Louvain**, is based on the exploration principle of the **Louvain** method. It consists in the optimization of the global criterion $QQ^{+}(\mathscr{P})$ defined by:

$$QQ^{+}(\mathscr{P}) = Q_{NG}(\mathscr{P}) + Q_{inertia}(\mathscr{P}) \tag{2}$$

with:

$$Q_{NG}(\mathscr{P}) = \frac{1}{2m} \Sigma_{vv'} \left[(A_{vv'} - \frac{k_v \cdot k_{v'}}{2m}) \delta(c_v, c_{v'}) \right] \tag{3}$$

where k_v is the degree of vertex $v \in V$, A is the adjacency matrix associated to G, m is the number of edges and δ the Kronecker function.

It may be noted that another combination of these criteria can be used, for instance to give more importance to one kind of data. However, in the general case where attributes

and relational information have the same weight, it is not useful to normalize the criteria $Q_{NG}(\mathscr{P})$ and $Q_{inertia}(\mathscr{P})$ because they have been normalized to take values between -1 and 1, as mentioned in the previous section.

The **I-Louvain** method is presented in Algorithm 1. The process begins with the discrete partition in which each vertex is in its own cluster (line 1). The algorithm is divided in two phases that are repeated.

ALGORITHM 1 : I-Louvain

Input : An attributed graph G
Output : A partition $\mathscr{P}_{res}$

1 $\mathscr{P} \leftarrow$ discrete partition of vertices of V;
2 $\mathcal{A} \leftarrow$ adjacency matrix of G;
3 $\mathcal{D} \leftarrow$ matrix of the squares of the euclidean distances between the vertices of V calculated on their attributes;
4 **repeat**
5 end $\leftarrow$ false;
6 $QQ^{+}_{anterior} \leftarrow QQ^{+}(\mathscr{P})$;
7 **repeat**
8 **foreach** *vertex* u *of* V **do**
9 $B \leftarrow$ neighbor community maximizing the gain of QQ^{+};
10 **if** *move of* u *in* B *induces a strictly positive gain* **then**
11 Affect u to the community B;
12 Update the partition $\mathscr{P}$ after the transfer of u into B;
13 **end**
14 **end**
15 **until** *no vertex can be moved anymore*;
16 **if** $QQ^{+}(\mathscr{P}) > QQ^{+}_{anterior}$ **then**
17 $G, \mathcal{A} \leftarrow$ Fusion_Matrix_Adjacency($\mathcal{A}$, $\mathscr{P}$);
18 $\mathcal{D} \leftarrow$ Fusion_Matrix_Inertia($\mathcal{D}$, $\mathscr{P}$);
19 **else**
20 end $\leftarrow$ true;
21 **end**
22 **until** *end*;
23 $\mathscr{P}_{res} \leftarrow \mathscr{P}$;

The first one is an iterative phase which consists in considering each vertex v and its neighbors in the graph and to evaluate the modularity gain induced by a move of v from its community to that of its neighbors. The vertex v is affected to the community for which the gain of the global criterion $QQ^{+}(\mathscr{P})$, defined in equation (2), is maximum. This process is applied repeatedly and sequentially for all vertices until no further improvement can be obtained.

If there is an increase of the modularity during the first phase, the second phase consists in building a new graph G' from the partition $\mathscr{P}'$ obtained at the end of the previous phase. This second phase involves two procedures: *Fusion_Matrix_Adjacency* and *Fusion_Matrix_Inertia*. The procedure *Fusion_Matrix_Adjacency* is identical to the

one used in the **Louvain** method [4] and it exploits only the relational information. It consists in building a new graph. The vertices of this new graph G' correspond to the communities obtained at the end of the previous phase. The weights of the edges between these new vertices are given by the sum of the weights of the edges between vertices in the corresponding two communities. The edges between vertices of the same community lead to a self-loop for this community in the new network.

The procedure *Fusion_Matrice_Inertia* exploits the attributes and allows to compute the distances between the vertices of G' from the distances between the vertices of G. If the graph G considered at the beginning of the iterative phase includes $|V|$ vertices then the matrix $\mathcal{D}$ is a symmetric square matrix of size $|V| \times |V|$ in which each term $\mathcal{D}[a, b]$ is the square of the distance between the vertices v_a and v_b of V. At the end of the iterative phase, a partition $\mathscr{P}'$ of V in k communities is obtained, in which each community will correspond to a vertex of V' in the new graph G' built by the procedure *Fusion_Matrix_Adjacency*. The matrix $\mathcal{D}'$ associated to this new graph G' is defined by:

$$\mathcal{D}'[x, y] = \sum_{(v_a, v_b) \in V \times V} \mathcal{D}[v_a, v_b] \cdot \delta(\tau(v_a), x) \cdot \delta(\tau(v_b), y) \tag{4}$$

where the function τ gives for each vertex $v \in V$ the vertex $v' \in V'$ corresponding to its cluster in $\mathscr{P}'$.

One advantage of the **Louvain** method is the local optimization of the modularity done during the first phase [2]. In the same way, in **I-Louvain**, the global modularity of a new partition can be quickly updated. There is no need to compute it again from scratch after each move of a vertex. Indeed, the modularity gain can be computed using only local information concerning the move of the vertex from its community to that of its neighbor. Given $\mathscr{P} = (A, B, C_1, .., C_r)$ the original partition and $\mathscr{P}' = (A \setminus \{u\}, B \cup \{u\}, C_1, .., C_r)$ the partition induced by the move of a vertex u from its community A to the community B where $A \setminus \{u\}$ denotes the community A deprived of the vertex u, the modularity gain induced by the transformation of $\mathscr{P}$ in $\mathscr{P}'$ is equal to:

$$\begin{aligned} \Delta Q_{inertia} =& Q_{inertia}(\mathscr{P}') - Q_{inertia}(\mathscr{P}) && (5) \\ =& \frac{1}{N \cdot I(V)} \sum_{v \in B} \left[\frac{I(V, u) \cdot I(V, v)}{2N \cdot I(V)} - D[u, v] \right] \\ & - \frac{1}{N \cdot I(V)} \sum_{v \in A \setminus \{u\}} \left[\frac{I(V, u) \cdot I(V, v)}{2N \cdot I(V)} - D[v, v'] \right] && (6) \end{aligned}$$

The proof of this proposition is not given due to the limited size of the article but it is detailed in [5]. One can notice that the variation of modularity resulting from the move of the vertex u from its community to an other one is the same whatever its new community. It follows that the modularity gain can be computed in taking only into account the increase (or decrease) induced by its affectation in its new community corresponding to the first term in Eq. 6. This confirms that the optimization of $Q_{inertia}$ can be done using a local computation based on the information related to the affectation of the vertex u in its new community.

5 Evaluation of I-Louvain method

Our first experiments aim at evaluating on a real dataset the performances of **I-Louvain**, which exploits attributes and relational data, compared with methods based only on one type of data, K-means for the attributes and Louvain for the relations and with ToTeM, an other community detection method designed for attributed graphs which exploits the two types of information, notably numerical attributes [6]. In the following experiments, we study the robustness of our method to various degradations of an artificial network and we compare its performances, according to the accuracy as well as the normalized mutual information, with K-means, Louvain and ToTeM. Among the methods exploiting the both kinds of data (relationships and attributes), Totem has been retained because it has been showned experimentally that it provides better results than simpler methods [6, 5] Finally, the last experiments aim at studying the impact of increasing the number of vertices and edges on the run-time evolution.

The **I-Louvain** source code and the dataset used for the experiments in the paper are available for download[3]. The **Louvain** source code is one proposed by Thomas Aynaud in 2009 [4].

5.1 Evaluation of I-Louvain method on a real network

Firstly, we present results obtained on a real dataset built using the databases DBLP (06/18/2014) and Microsoft Academic Search (02/03/2014). DBLP allows to generate a graph $G = (V, E)$ with $|V|$ = 2515 and $|E|$ = 5313 that reflects the coauthor relationship: a vertex represents an author and two authors are linked if they have copublished at least one article in a conference in computer science also refereed in Microsoft Academic Search. The 23 keywords (data mining, Computer vision, etc.) associated to the conferences in the Microsoft Academic Search database are used to define 23 attributes on the vertices: the number of publications of an author in conferences associated to a given keyword corresponds to a component of his attribute vector. These keywords allow also to define a partition corresponding to the ground truth for this dataset: the true community of an author corresponds to the research field, identified by the corresponding key word, in which he has mainly published.

The results are evaluated using the Normalized Mutual Information (NMI) derived from the mutual information (MI) and entropy (H), and defined by :[33]

$$NMI(\mathscr{P}_1, \mathscr{P}_2) = \frac{MI(\mathscr{P}_1, \mathscr{P}_2)}{\sqrt{H(\mathscr{P}_1)H(\mathscr{P}_2)}} \tag{7}$$

Table 1 presents the results provided by I-Louvain and those obtained by Louvain, K-means with $K = 22$ and ToTem. In this experiment, where we have a ground truth, the results confirm the interest of using the two kinds of information. Indeed the NMI of K-means is equal to 0.58 whereas the number of clusters that must be identified is given as parameter for this algorithm, when it is equal to 0.69 for Louvain. Moreover,

[3] I-Louvain source code and dataset: `http://bit.ly/ILouvain`

[4] http://perso.crans.org/aynaud/communities/

with a NMI equals to 0.72, the proposed method outperforms ToTeM which obtains only 0.69. These results confirm the interest of I-Louvain to improve the detection of the communities.

Table 1. Evaluation according to the normalized mutual information (NMI)

	Louvain	K-means	ToTeM	I-Louvain
NMI	0.69	0.58	0.69	**0.72**

5.2 Evaluation of I-Louvain method on artificial data

In this second set of experiments, we evaluate the robustness of our method on artificial networks after different transformations of a reference network R, composed of 168 edges and 99 vertices uniformly distributed into 3 classes. This reference network has also been generated with the model proposed by Dang [8]. Moreover, each vertex is described by an attribute following a normal distribution with a standard deviation σ equal to 7 and a mean equal to $m_1 = 10$ for the first class, $m_2 = 40$ for the second class and $m_3 = 70$ for the third class. The class of the vertex in R is used as a ground truth for the evaluation. From this reference network R we built four families of networks:

- R.1.x in which the relational information is weakened in R, by the substitution of a percentage p of edges within class by edges between classes with $p = 0.25$ for R.1.1 and $p = 0.5$ for R.1.2;
- R.2.x in which the values of the attributes are less representative of each class, with a standard deviation $\sigma = 10$ for R.2.1 and $\sigma = 12$ for R.2.2;
- R.3.x which contain more vertices than R, 999 vertices for R.3.1 and 5,001 for R.3.2;
- R.4.x which contain more edges than R by introducing respectively 5 edges per new vertex in R.4.1 and 10 in R.4.2.

The results of **I-Louvain** are compared to those of the **Louvain** method, **K-means** with k = 3 and **ToTeM**. Tables 2 and 3 present respectively the accuracy (AC) and normalized mutual information (NMI). In exploiting the attributes and the relational information, the **I-Louvain** method is more robust than the **Louvain** method in the case of a degradation of the relational information. The **K-means** gives good results when the size of the network increases, but it requires the number of clusters as parameter. Despite this advantage, it obtains less good results than **I-Louvain** in front of a degradation of the attributes, notably for the NMI. Finally, compared to **ToTeM**, **I-Louvain** produces better or similar results. It is notably better for a larger number of vertices.

5.3 Run-time of I-Louvain

In the last set of experiments, we evaluate the run-time of **I-Louvain** on different networks. **Figure 1** presents the run-time evolution against the number of vertices $|V|$. In our experiments, we consider attributed networks with two attributes and where the number of edges $|E| = 3 \times |V|$. These results indicate that **I-Louvain** is able to handle large graphs.

Table 2. Evaluation according to the accuracy (AC) and the number of clusters (#cl) (* means that the transformation has no influence on the results for this method)

	Louvain		K-means	ToTeM		I-Louvain	
	AC	#cl.	AC	AC	#cl.	AC	#cl.
Reference network							
R	84%	4	96%	97%	3	**98%**	3
Degradation of the relational information							
R.1.1	33%	8	96%*	18%	30	**78%**	5
R.1.2	23%	9	96%*	14%	36	**63%**	6
Degradation of the attributes							
R.2.1	84%*		90%	95%	3	**96%**	3
R.2.2	84%*		87%	20%	26	**98%**	3
Number of vertices							
R.3.1	50%	11	**97%**	**97%**	3	84%	4
R.3.2	40%	12	**98%**	0,5%	1,518	85%	4
Number of edges							
R.4.1	**96%**	3	96%*	95%	3	94%	3
R.4.2	97%	3	96%*	**98%**	3	**98%**	3

Table 3. Evaluation according to the NMI (* means that the transformation has no influence on the results for this method)

NMI	Louvain	K-means	ToTeM	I-Louvain
Reference network				
R	0.78	0.88	0.86	**0.93**
Degradation of the relational information				
R.1.1	0.22	0.88*	0.48	**0.60**
R.1.2	0.11	0.88*	**0.37**	0.35
Degradation of the attributes				
R.2.1	0.78*	0.72	0.81	**0.88**
R.2.2	0.78*	0.63	0.56	**0.93**
Number of vertices				
R.3.1	0.59	**0.88**	0.85	0.80
R.3.2	0.58	**0.89**	0.37	0.77
Number of edges				
R.4.1	**0.84**	0.88*	0.80	0.81
R.4.2	0.87	0.88*	**0.91**	**0.91**

6 Conclusion

In this article, we studied the problem of attributed graph clustering when the vertices are described by real attributes. Inspired by the Newman modularity, we introduce a modularity measure, based on inertia. This measure is suited for assessing the quality of a partition of elements represented in a real vector space. We also introduced

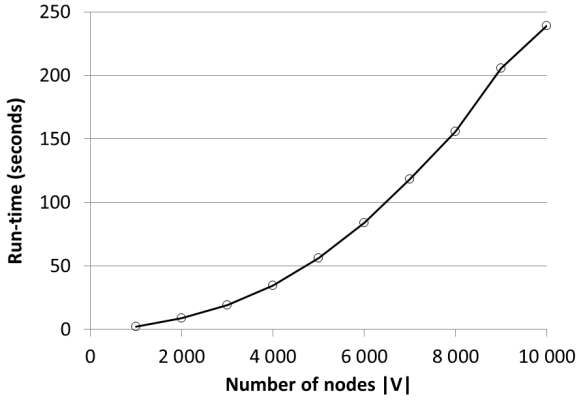

Fig. 1. Run-time of **I-Louvain** on different networks $G = (V, E)$ with $|E| = 3 \times |V|$

I-Louvain, an algorithm which combines our criterion with Newman's modularity in order to detect communities in attributed graphs. We demonstrated formally that this new algorithm can be optimized in its iterative phase. As we show in the experiments, using jointly the relational information and the attributes, **I-Louvain** detects more efficiently the communities than ToTeM or methods using only one type of data. Moreover, the method is resistant toward a degradation of the relations or the attributes, an increase in the density of the relations or the size of the network. Finally, the experiments confirm the scalability of the method.

The authors would like to thank P.N. Mougel for his help in building the bibliographic dataset.

References

1. Arenas, A., Fernández, A., Gómez, S.: Analysis of the structure of complex networks at different resolution levels. New Journal of Physics 10(5), 053039 (2008)
2. Aynaud, T., Blondel, V., Guillaume, J.L., Lambiotte, R.: Multilevel local optimization of modularity. In: Graph Partitioning. pp. 315–345. Wiley (2013)
3. Bichot, C., Siarry, P.: Graph partitioning. Wiley (2013)
4. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of community hierarchies in large networks. CoRR abs/0803.0476 (2008)
5. Combe, D.: Detection de communautes dans les reseaux d'information utilisant liens et attributs,. In: PhD Thesis, Jean Monnet University, Lyon (2013)
6. Combe, D., Largeron, C., Egyed-Zsigmond, E., Géry, M.: Combining relations and text in scientific network clustering. In: International Conference on Advances in Social Networks Analysis and Mining (ASONAM). pp. 1280–1285 (2012)
7. Cruz, J.D., Bothorel, C., Poulet, F.: Entropy based community detection in augmented social networks. In: Computational Aspects of Social Networks (CASoN 2011). pp. 163–168 (2011)
8. Dang, T.A.: Analysis of community in social networks. In: PhD Thesis, Paris 13 (2013)

9. Dang, T.A., Viennet, E.: Community Detection based on Structural and Attribute Similarities. In: International Conference on Digital Society (ICDS). pp. 7–12 (2012)
10. Ding, C., He, X., Zha, H., Gu, M.: A min-max cut algorithm for graph partitioning and data clustering. In: IEEE International Conference on Data Mining. pp. 107 – 114 (2001)
11. Elhadi, H., Agam, G.: Structure and attributes community detection: Comparative analysis of composite, ensemble and selection methods. In: 7th Workshop on Social Network Mining and Analysis. pp. 10:1–10:7. SNAKDD '13, ACM, New York, NY, USA (2013)
12. Ester, M., Ge, R., Gao, B., Hu, Z., Ben-Moshe, B.: Joint Cluster Analysis of Attribute Data and Relationship Data: the Connected k-Center Problem. In: SIAM International Conference on Data Mining. pp. 25–46. ACM Press (2006)
13. Fjällström, P.O.: Algorithms for graph partitioning: a survey. Science 3(10) (1998)
14. Flake, G., Tarjan, R., Tsioutsiouliklis, K.: Graph clustering and minimum cut trees. Internet Mathematics 1(4), 385–408 (2003)
15. Fortunato, S.: Community detection in graphs. Physics Reports 486(3-5), 75–174 (Jun 2010)
16. Ge, R., Ester, M., Gao, B.J., Hu, Z., Bhattacharya, B., Ben-Moshe, B.: Joint cluster analysis of attribute data and relationship data. ACM Transactions on Knowledge Discovery from Data 2(2), 1–35 (2008)
17. Günnemann, S., Farber, I., Boden, B., Seidl, T.: Subspace Clustering Meets Dense Subgraph Mining: A Synthesis of Two Paradigms. In: IEEE International Conference on Data Mining. pp. 845–850 (2010)
18. Günnemann, S., Boden, B., Seidl, T.: DB-CSC: a density-based approach for subspace clustering in graphs with feature vectors. Machine Learning and Knowledge Discovery in Databases pp. 565–580 (2011)
19. Kernighan, B.W., Lin, S.: An Efficient Heuristic Procedure for Partitioning Graphs. Bell System Technical Journal 49(2), 291–307 (1970)
20. Lancichinetti, A., Fortunato, S.: Community detection algorithms: a comparative analysis. Physical review E 80(5), 056117 (2009)
21. Li, H., Nie, Z., Lee, W.C.W., Giles, C.L., Wen, J.R.: Scalable Community Discovery on Textual Data with Relations. 17th ACM conference on Information and knowledge management pp. 1203–1212 (2008)
22. Moser, F., Ge, R., Ester, M.: Joint Cluster Analysis of Attribute and Relationship Data Without A-Priori Specification of the Number of Clusters. In: 13th ACM SIGKDD conference on knowledge discovery and data mining. pp. 510–519 (2007)
23. Newman, M.: Detecting community structure in networks. The European Physical Journal B-Condensed Matter and Complex Systems 38(2), 321–330 (2004)
24. Newman, M.: Modularity and community structure in networks. Proceedings of the National Academy of Sciences of the United States of America 103(23), 8577–8696 (2006)
25. Newman, M., Girvan, M.: Finding and evaluating community structure in networks. Physical review E 69(2), 1–16 (2004)
26. Prado, A., Plantevit, M., Robardet, C., Boulicaut, J.F.: Mining graph topological patterns: Finding covariations among vertex descriptors. IEEE Trans. Knowl. Data Eng. 25(9), 2090–2104 (2013)
27. Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. Physical Review E 74(1), 016110 (2006)
28. Satuluri, V., Parthasarathy, S.: Scalable graph clustering using stochastic flows: applications to community discovery. In: 15th SIGKDD conference on Knowledge discovery and data mining. pp. 737–746 (2009)
29. Schaeffer, S.: Graph clustering. Computer Science Review 1(1), 27–64 (2007)
30. Shi, J., Malik, J.: Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22(8), 888–905 (2000)

31. Stattner, E., Collard, M.: From frequent features to frequent social links. International Journal of Information System Modeling and Design (IJISMD) 4(3), 76–98 (2013)
32. Steinhaeuser, K., Chawla, N.: Community detection in a large real-world social network. Social Computing, Behavioral Modeling, and Prediction pp. 168–175 (2008)
33. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research 3, 583617 (2003)
34. Von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)
35. Yang, J., McAuley, J.J., Leskovec, J.: Community detection in networks with node attributes. In: ICDM. pp. 1151–1156 (2013)
36. Zhou, Y., Cheng, H., Yu, J.: Graph clustering based on structural/attribute similarities. VLDB Endowment 2(1), 718–729 (2009)
37. Zhou, Y., Cheng, H., Yu, J.X.: Clustering Large Attributed Graphs: An Efficient Incremental Approach. 2010 IEEE International Conference on Data Mining pp. 689–698 (2010)